

**UNIVERSIDAD MILITAR
NUEVA GRANADA**



**DESARROLLO DE UNA APLICACIÓN MULTIJUGADOR ONLINE DIRIGIDA
POR UN MICRÓFONO**

NESTOR AUGUSTO GOYES ORTEGA

Tesis

Director
Carlos Aranzazu López
Diseñador Gráfico

**UNIVERSIDAD MILITAR NUEVA GRANADA
FACULTAD DE INGENIERÍA
INGENIERÍA EN MULTIMEDIA
BOGOTÁ
2012**

**DESARROLLO DE UNA APLICACIÓN MULTIJUGADOR ONLINE DIRIGIDA
POR UN MICRÓFONO**

NESTOR AUGUSTO GOYES ORTEGA

**UNIVERSIDAD MILITAR NUEVA GRANADA
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA EN MULTIMEDIA
BOGOTÁ D.C.
2012**

**DESARROLLO DE UNA APLICACIÓN MULTIJUGADOR ONLINE DIRIGIDA
POR UN MICRÓFONO**

NESTOR AUGUSTO GOYES ORTEGA

**Director De Proyecto:
CARLOS ARANZAZU**

INFORME FINAL DE PROYECTO DE GRADO

**UNIVERSIDAD MILITAR NUEVA GRANADA
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA EN MULTIMEDIA
BOGOTÁ D.C.
2012**

TABLA DE CONTENIDO

	Pág.
INTRODUCCIÓN	10
1. OBJETIVOS.....	11
1.1 OBJETIVO GENERAL:	11
1.2. OBJETIVOS ESPECÍFICOS:.....	11
2. ANTECEDENTES.....	12
3. JUSTIFICACIÓN.....	15
4. MARCO TEÓRICO	16
4.1 LOS VIDEOJUEGOS:.....	16
4.2 LA WEB 2.0	17
4.3 FUNCIONAMIENTO DE ARQUITECTURAS CLIENTE-SERVIDOR.....	18
4.4 SMARTFOX SERVER	19
4.5 ACTION SCRIPT	20
4.6 CONEXIONES CON BASES DE DATOS.....	21
5. METODOLOGÍA	22
5.1 ANÁLISIS.....	23
5.1.1. Modelo Conceptual.	23
5.1.1.1. Ambiente a comunicar.	23
5.1.1.2 Requerimientos de la aplicación	23
5.1.2. Modelo estructural.	23
5.1.2.1. Funciones del juego.....	23
5.1.2.2 Herramientas a utilizar.	24

5.1.3 Modelo Organizacional.	24
5.1.3.1 Garaje.	24
5.1.3.2 Puntajes propios.	24
5.1.3.3 Puntajes generales o Ranking.	24
5.1.3.4 Creación de Salas de Juego	24
5.2 DISEÑO DE LA ESTRUCTURA	25
5.2.1 Casos de Uso.	25
5.2.3 Estructuración de la base de datos	30
5.3 DISEÑO GRÁFICO	31
5.3.1 Nombre de la aplicación.....	31
5.3.2 Logotipo.	33
5.3.2.1 Deportes Automovilísticos.....	33
5.3.2.2 Controles del juego.	33
5.3.2.3 Personaje Principal Diseñado.....	33
5.3.3 Diseño de los automóviles.	35
5.3.4 Escenarios.	36
5.3.5 Interfaz.....	37
5.3.5.1 Comportamiento.	37
5.3.5.2 Presentación.....	37
5.3.5.3 Asistencia al usuario.	41
5.4 IMPLEMENTACION.....	42
5.4.1 Configuración de Smartfox Server local.....	42
5.4.2 La captura de audio.	43
5.4.3 Registro de jugadores nuevos y puntajes.	43

5.4.4 Ingreso de jugadores registrados.....	44
5.4.5 Creación de salas de chat.....	44
5.4.5 Creación de salas de juego y carreras multijugador	44
5.4.6 Instalación de Smartfox en servidor real e IP fija.	45
6. EVALUACION DEL PRODUCTO	47
6.1. PRUEBAS DE FUNCIONALIDAD.....	48
6.2 PRUEBAS DE USABILIDAD.....	49
7. RESULTADOS.....	51
7.1. RESULTADOS DE LAS PRUEBAS	51
7.2 CAMBIOS EN BASE A PRUEBAS	51
7.2.1 Correcciones Inmediatas	51
7.2.2 Mejoras para la segunda versión.	53
7.3 RESULTADOS DEL DESARROLLO	54
8. POSIBLES VÍAS DE COMERCIALIZACION DE LA APLICACION	55
8.1 EVENTOS ENTRE CENTROS COMERCIALES	56
8.2 CONCURSOS VIA TELEFONICA EN TELEVISION	56
8.3 PAUTAS DE MARCAS AUTOMOTRICES.....	56
CONCLUSIONES	57
BIBLIOGRAFIA.....	58

LISTA DE FIGURAS

	Pág.
Figura 1. Chamber of chat, banner publicitario	12
Figura 2. Sala de chat club penguin.....	13
Figura 3 Pantalla de bienvenida.....	13
Figura 4. Pantalla de bienvenido y pistas Zoom	14
Figura 5. Transferencia de información entre el servidor y diferentes clientes	18
Figura 6. Arquitectura cliente-servidor	19
Figura 7. Tecnologías que soporta Smartfox Server.....	20
Figura 8. Diagrama de Casos de Uso Lobby	25
Figura 9. Casos de uso Registro de nuevos usuarios.....	26
Figura 10. Casos de uso Chat en zona principal	26
Figura 11. Casos de uso Carrera.....	27
Figura 12. Diagrama de clases SmartFox Server	29
Figura 13. Diagrama de Clases PHP	30
Figura 14. Diagrama De La Base De Datos.....	31
Figura 15. Bocetos del casco y cuerpo del personaje.....	32
Figura 16. Personaje principal vectorizado	32
Figura 17. Boceto del megáfono a incluir.....	33
Figura 18. Propuesta logo 1	35
Figura 19. Propuesta logo 2.....	34
Figura 20. Propuesta logo 3.....	34
Figura 21. Propuesta seleccionada y vectorizada.....	35
Figura 22. Automóviles escogidos y su respectiva caricatura.....	35
Figura 23. Bocetos de pistas.....	36
Figura 24. Vectorizacion de pistas	37
Figura 25. Diagramación de la interfaz principal	38
Figura 26. Propuesta de diagramación de las carreras	38
Figura 27. Apariencia de la interfaz principal	39

Figura 28. Diseño de las ventanas emergentes.....	39
Figura 29. Apariencia de las carreras	40
Figura 30. Screen shot de config.xml.....	43
Figura 31. Diagrama de funcionamiento del registro de usuarios	44
Figura 32. Screen shot del panel de control de smartfox.....	45
Figura 33. Screen shot de una carrera en juego.....	45
Figura 34. Diagrama de funcionamiento usuario-hosting-smartfox.....	46

LISTA DE TABLAS

	Pág.
Tabla 1. Posibles vías de comercialización de la aplicación	55

INTRODUCCIÓN

En Internet podemos encontrar un sinnúmero de aplicaciones y juegos de deportes, estrategia y de muchos otros géneros que cumplen con la función de entretener al usuario regular, permitiendo que jugadores de todo el mundo interactúen con otros desde sus casas; Sin embargo, después de ver portales dedicados 100% a los minijuegos en línea, se hace necesario que este tipo de videojuegos ofrezcan interacción mediante otros medios como el audio, y se vuelvan plataformas completas para los usuarios, permitiéndoles desempeñar un rol mucho más amplio dentro de una aplicación, ya sea solo o multijugador.

Este documento describe desde el punto de vista teórico las aplicaciones web 2.0 y el funcionamiento de arquitecturas cliente-servidor. Por otro lado, desde el punto metodológico se describe el desarrollo de una aplicación multijugador basado en las etapas de desarrollo de software que propone Roger Pressman ¹ y consiste básicamente en el análisis de requerimientos y contenido, el diseño y la implementación de la aplicación.

En el desarrollo de este proyecto podremos apreciar la importancia de fundamentar las aplicaciones web en el rol social que cada uno de los usuarios desempeña en la misma, además de usar hardware diferente al usual para controlar objetos dentro de un juego online.

¹ PRESSMAN, Roger. Ingeniería de Software. Un enfoque práctico. 5 ed. Lugar: McGraw Hill, 2002. 900p. ISBN

1. OBJETIVOS

1.1 OBJETIVO GENERAL:

Desarrollar una aplicación multiusuario, que se desempeñe en el área del entretenimiento online, y se controle parcialmente con captura de audio, acelerando carros de carreras con el nivel de la voz.

1.2. OBJETIVOS ESPECÍFICOS:

- Implementar un sistema de captura de audio que mida el nivel de voz del jugador para ser usado en la aceleración de objetos vectoriales graficados como automóviles de carreras.
- Diseñar una interfaz gráfica con un nivel de usabilidad suficiente para que el usuario comprenda el contenido y la estructura de la aplicación propuesta, basando su desarrollo en métodos de medición de usabilidad eficaces.
- Crear un entorno de juego que se actualice simultáneamente en los ordenadores de los usuarios activos en las salas de juego, para lograr una experiencia multijugador ágil y confiable.

2. ANTECEDENTES

La red mundial cuenta con espacios completamente dedicados a los juegos online, que nos ofrecen minijuegos de infinitas categorías como deportes, peleas, aventura, estrategia, azar, agilidad, entre otros. Portales destacados como JUEGOS.COM o MINIJUEGOS.COM son sitios actualizados casi a diario que nos dejan probar los más recientes juegos en línea y son portales que permiten a los usuarios jugarlos completamente gratis.

Trasladándonos a gran escala, podemos encontrar los portales dedicados a un solo juego en línea como es el caso de FOOPETS.COM el cual nos da la oportunidad de adoptar nuestra propia mascota, alimentarla, cuidarla y jugar con ella, siendo un juego más completo y que deja al usuario ser parte activa y constante del sitio web, pero en este formato de videojuegos en línea es clave destacar para el presente proyecto las aplicaciones multijugador que se han creado con tecnologías como SMARTFOX SERVER ², que además de ser reconocidas internacionalmente envuelven al jugador de una manera única, a continuación describiré algunas de ellas:

En la figura 1, podemos ver un corte gráfico de Chamber of Chat, una aplicación multiusuario que ofrece una interfaz completamente nueva a los visitantes, que les permite chatear diseñando su propio avatar de la saga HARRY POTTER, sumergiéndolo en una nueva experiencia tridimensional para chatear. Recomendada por Adobe desde 2007.

Figura 1. Chamber of chat, banner publicitario



Fuente: www.chamberofchat.com

Club Penguin es uno de los portales de chat más conocidos del mundo, desarrollado por Disney y dirigido a los niños desde los 10 años, que cuenta con un entorno amistoso y permite a los usuarios tener su propio pingüino, viajar a

² Smartfox Server Official Site. Smartfox Showcase. (citado en febrero 5 de 2011). Disponible en Internet www.smartfoxserver.com/showcase

diferentes lugares del polo norte y divertirse chateando con los demás pingüinos (usuarios) se puede visualizar su interfaz gráfica en la Figura No. 2.

Figura 2. Sala de chat club penguin



Fuente: www.clubpenguin.com

Petpet Par es una aplicación lanzada por Nickelodeon en el año 2009 que permite a los niños desde los 11 años de edad interactuar con otros y jugar minijuegos juntos dentro del mismo portal. Petpet Park es un mundo de fantasía y ambientes distintos cada vez, que ofrece numerosas experiencias y actividades que hacen del juego una actividad interminable a propósito y una excelente opción para pasar el tiempo libre. Es posible apreciar la pantalla de bienvenida de esta aplicación en la Figura No. 3.

Figura 3 Pantalla de bienvenida



Fuente: www.petpetparkr.com

Una de las aplicaciones más similares a la que se desarrolla en este proyecto, es el juego Zoom Zoom (Figura 4), enfocada a dispositivos móviles como el Iphone y el Ipad. Los autos son acelerados por la voz y el usuario puede correr en diferentes pistas.

Figura 4. Pantalla de bienvenida y pistas Zoom



Fuente: Apple Store.

En el ámbito institucional y entre los proyectos de grado desarrollados en la Universidad Militar Nueva Granada no hay muchas aplicaciones multijugador, siendo un área poco experimentada localmente, así que deja mucho que aportar una aplicación multiusuario comandada por niveles de voz.

3. JUSTIFICACIÓN

La exigencia del Usuario hoy en día, lleva al desarrollador a crear una experiencia ágil, innovadora, divertida y rápida para los jugadores de un portal. Dando lugar a la evolución de los mandos que emplea un usuario para jugar online, y al papel que desempeñan los mismos dentro de la aplicación.

En cuanto a lo social, la necesidad de entretenimiento que posee una sociedad como la nuestra, pide cada día nuevos sitios para visitar y pasar el tiempo libre.

Dentro de los proyectos de grado realizados con respecto a la Internet en los últimos años, no se ha implementado ningún juego online de carácter multijugador en la Universidad Militar Nueva Granada, por lo cual es un campo nuevo a profundizar que me permitirá compartir conocimientos básicos sobre tecnologías socket, el tiempo real y la programación de aplicaciones robustas en Action Script, aportando fundamentos para la posterior consulta de los mismos por parte de otros compañeros; además, una aplicación como la que se desarrollará puede llegar a traer beneficios de carácter económico al ser una aplicación innovadora y cautivante para los usuarios y seguramente atractiva para compañías.

4. MARCO TEÓRICO

4.1 LOS VIDEOJUEGOS:

“Los procesos Lúdicos, y en especial los juegos, son senderos abiertos a la creatividad y al conocimiento...”³

“Un juego está compuesto por reglas, ambiente, obstáculos, objetivos, y el nombre que es lo que identifica a un juego...”⁴

Teniendo en cuenta la anterior afirmación, podemos dividir el juego de video en 5 partes fundamentales que permiten clasificar, jugar e identificarlo dentro de un género o clase en específico:

Empezando por lo fundamental, LAS REGLAS de un juego son las que marcan pauta para jugarlo, son cada una de las normas que el jugador debe cumplir, que no sólo presentan al juego sino que orientan al jugador hacia los objetivos clave de la actividad. Por otro lado, EL AMBIENTE que ofrece un videojuego, permite al jugador contextualizarse en un lugar o situación específica, lo cual lleva el juego a contar e intentar resolver un problema que se presenta en un lugar y tiempo determinado, dando lugar a guiones que cuentan historias o que simplemente envuelven al jugador en actividades simples, pero divertidas

La actividad que desempeña un jugador siempre debe tener un ¿Por Qué?, que se descompone en LOS OBJETIVOS del juego, lo que busca el jugador y la meta que quiere alcanzar en cada uno de los aspectos de la historia o el problema presentado. De los objetivos nace el ¿COMO alcanzarlos? El cual está delimitado por las reglas y obstáculos del juego, y explotado por la habilidad del jugador. Por último, el nombre del videojuego permite diferenciarlo de otros de su género y crea una identidad propia del mismo, debe estar relacionado con el ambiente, los objetivos y obstáculos del juego.

Atributos como los anteriormente descritos son tenidos en cuenta para la realización de un juego de video completo, pero ¿Qué tal implementar un juego para la Web en donde puedan interactuar varios jugadores a la vez?

³ JIMÉNEZ VÉLEZ, Carlos Alberto. El juego, Nuevas miradas desde la Neuropedagogía. Magisterio. 2008.

⁴ MEDINA, Juan David. Desarrollo de aplicación Online para manejo de clientes. Universidad Militar Nueva Granada. 2008

4.2 LA WEB 2.0

Al pasar la última década, la Web ha tomado otro rumbo, tanto para los desarrolladores como para los usuarios de la misma, teniendo en cuenta que el usuario, hace cada vez mas parte activa de las aplicaciones que se encuentran en la Internet, respecto a el hardware que usa, el rol que desempeña, y la variedad que le ofrezcan a diario; por lo tanto la llamada Web 2.0 se adueña de la Red día a día.⁵

Una aplicación Web 2.0 se compone de algunas áreas fundamentales, como las hojas de estilo, lenguajes y estándares. A continuación nombraré algunos componentes fundamentales de una aplicación 2.0:

Desde los años 90's se empezaron a trabajar los estándares W3C, manejados por la empresa del mismo nombre y que nos indican cuales son las novedades y cuáles son los requerimientos que debe cumplir un sitio o una aplicación para ser certificada, la manera correcta de usar xml, html, Xhtml, Javascript, entre otros lenguajes usados en la Web⁶. Por otro lado el diseño de interfaz y de la plataforma, es clave para que una aplicación sea 2.0, es necesario el uso adecuado de las hojas de estilo, incluir contenido RSS y por supuesto, compilar sectores Flash o Flex, ofreciendo al usuario un mayor dinamismo en cualquier portal o aplicación desarrollada en navegadores o en el mismo escritorio del ordenador.

El factor social es muy importante en las aplicaciones ya que obligan al usuario a ser parte activa de la misma, aportando contenido e interactuando con diferentes usuarios ya sea en una plataforma de uso individual o de uso múltiple. En esta oportunidad estoy enfocado hacia las aplicaciones con múltiples usuarios simultáneos, así que, iniciando con este tema, podemos definir una aplicación multiusuario como un conjunto de factores que ofrecen una experiencia paralela a dos o más usuarios, permitiendo que cada uno de ellos interactúe con un entorno de manera simultánea, siempre con un objetivo en específico.⁷

Un multijugador online, implica que varios jugadores estén conectados desde diferentes ordenadores con acceso a internet, los cuales reciben la actualización del entorno constantemente desde un servidor con la capacidad para ello.

⁵ Baxley Bob. *Making the web work; Designing effective web applications*. Sams. 2002.

⁶ Estándares W3C. Web design applications. <http://www.w3c.es/estandares/>

⁷ Yue Yang, Changjia Chen, Yishuai Chen. The Study of Online Games Based on Virtual Social Relationship. Beijing Jiaotong University. 2009. IEEE Article.

4.3 FUNCIONAMIENTO DE ARQUITECTURAS CLIENTE-SERVIDOR

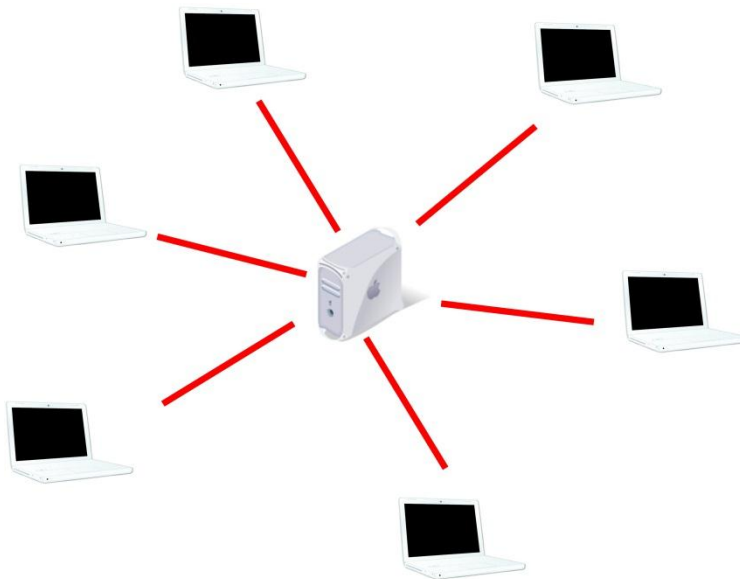
La tecnología Cliente-Servidor es la relación entre procesos que corren en máquinas separadas dando lugar a un intercambio de información constante entre ellas, una de las máquinas que componen esta relación es denominada **CLIENTE**, la cual envía peticiones de información o servicios; es la consumidora principal del servicio.

EL SERVIDOR es el ordenador o recurso que está totalmente dedicado a responder los requerimientos del cliente, en resumen es el proveedor de los servicios que necesitan los clientes, conectados al servidor por medio de redes de acceso público o privado, según la utilidad que se le dé a la arquitectura.⁸

Frecuentemente el servidor necesita correr aplicaciones diferentes según la clase de petición de los clientes, para lo cual han nacido lenguajes y sistemas capaces de devolver al cliente lo que pide, ya sea contenido informativo, actualización de entornos, entre otros...⁹

En la Figura 5 podemos ver de manera gráfica la manera cómo funciona el intercambio de información entre los diferentes clientes y el mismo servidor.

Figura 5. Transferencia de información entre el servidor y diferentes clientes

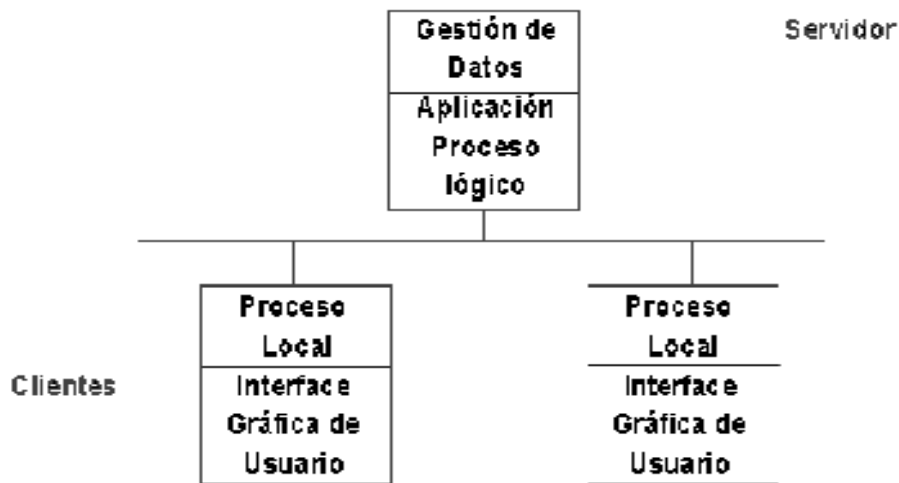


Fuente: Autor

⁸ Ahmed Abdelkhalek. Parallelization and Performance of Interactive Multiplayer Game Servers. University of Toronto. 2004. IEEE Article

⁹ José Guillermo Valle. Arquitectura Cliente-Servidor, Monografía. Corporación Universitaria Minuto de Dios. Colombia, 2005.

Figura 6. Arquitectura cliente-servidor



Fuente: VALLE, José Guillermo. Arquitectura cliente-servidor. Monografía Corporación Universitaria Minuto de Dios. Colombia, 2005.

Al tocar el tema de las aplicaciones cliente/servidor es claro que necesitamos una manera de conectar el proceso que se desempeña en una aplicación del lado del cliente, con el proceso que desempeñará una aplicación del lado del servidor para devolver la información pedida (Figura 6), a estas conexiones o “enchufes” se les conoce con el nombre de SOCKETS, los cuales comunican un programa del cliente con otro programa del servidor de una red, ya que hacen parte de los puntos finales de conexión en cuanto a transferencia de información se refiere.¹⁰

4.4 SMARTFOX SERVER

En el caso del juego a desarrollar, se implementará la tecnología CLIENTE-SERVIDOR ofrecida por SmartFox server, una aplicación basada en sockets, que permite al desarrollador crear aplicaciones de carácter multiusuario, sin necesidad de recurrir a programación del lado del servidor, ya que cuenta con los recursos suficientes para ofrecer un óptimo desempeño de las aplicaciones desarrolladas. Además soporta diferentes lenguajes y tecnologías como Java o plataformas Android (Figura 7).

¹⁰ Ibid. Pg. 50

Figura 7. Tecnologías que soporta Smartfox Server



Fuente: www.smartfoxserver.com¹¹

Smartfox server permite al desarrollador usar la librería Action Script incluida en el paquete de la aplicación, para agilizar el proceso de programación de un juego, la cual contiene clases referentes a la actualización de datos, intercambio de mensajes, Pop ups, entre otras utilidades.

Existen otras aplicaciones como Red 5 que intentan llenar las expectativas del desarrollador, sin embargo al ser una aplicación open source se encuentra en constante cambio aun, cuenta con pocos ejemplos y referencias, además de poco respaldo. Pero contando con librerías de la magnitud de Smartfox server es cercano pensar en aplicaciones robustas, funcionales rápidas y confiables, que reducen riesgos en cuanto a su aplicabilidad y costos comparándolo con tecnologías como Flash Media Server, además facilita su entendimiento gracias a la documentación, los ejemplos que ofrece, y lo más atractivo, está a la disposición de Action Script, un lenguaje poderoso y relativamente sencillo de implementar, el cual se describe a continuación.

4.5 ACTION SCRIPT

Action Script es un lenguaje de programación que desde sus inicios ha estado enfocado en dar comandos a objetos vectoriales creados en Adobe Flash; con una arquitectura y sintaxis similar a la de java, podemos resaltar Action Script como uno de los lenguajes más potentes y conocidos en el ámbito de aplicaciones Online. Hay que tener en cuenta que al trabajar en aplicaciones que integran diferentes herramientas y librerías, debemos analizar cuáles de ellas nos traen más ventajas al usarlas.

Actualmente se usan dos versiones diferentes del lenguaje Action Script, la 2.0 y 3.0.

¹¹ Smartfox Server Official Site. Smartfox Showcase. . <http://www.smartfoxserver.com/showcase>

ACTION SCRIPT 2.0. Un lenguaje basado en pequeños scripts, enfocado principalmente en diseñadores que requieren de cierto grado de programación para sus proyectos, como páginas web, multimedios educativas, entre otros. Sin embargo Action Script 2.0 también puede ser usado por desarrolladores de un nivel más alto, que requieren orientar a objetos sus proyectos y definen clases y funciones propias.

ACTION SCRIPT 3.0. Un lenguaje totalmente orientado a objetos, la sintaxis cambian relevantemente en ésta versión del lenguaje y exige al desarrollador la definición de funciones e implementación de eventos de manera más organizada, dando agilidad a los procesos que lleva a cabo la aplicación desarrollada.

Action Script 3.0 también facilita el manejo de espacios 3D dentro de un lienzo 2D como el que maneja el escenario de Adobe Flash.

En el desarrollo de esta aplicación multijugador se usará la versión 2.0 de éste lenguaje, ya que las librerías de SmartFox server ofrecen compatibilidad completa con Action Script 2.0 y bases de datos MySQL, mientras que el desarrollo de las mismas para Action Script 3.0 aún se encuentra en progreso, a pesar de que ya están en el mercado.

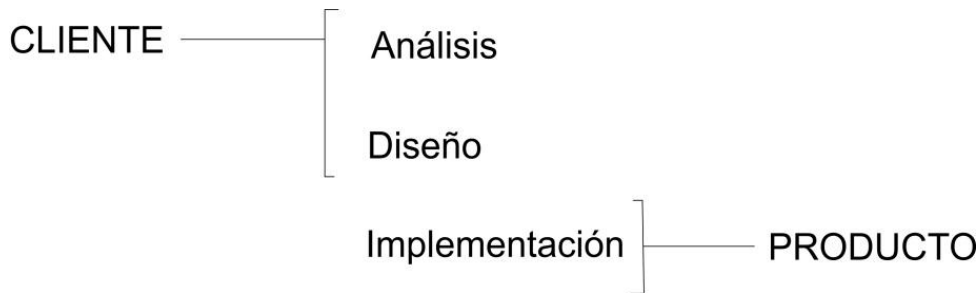
4.6 CONEXIONES CON BASES DE DATOS

La aplicación a desarrollar requiere que sus usuarios se autentiquen antes de usarla, por lo cual deben estar registrados en una base de datos que recoge su información básica, además de sus atributos dentro del juego, como su puntaje, automóvil y garaje. Pero el verdadero asunto no es almacenar esta información dentro de la base de datos, sino conectarla con una aplicación desarrollada en Action Script. Para esto es útil usar lenguajes como PHP 5, un lenguaje de programación web que es interpretado del lado el servidor y desde el cual se puede implementar de manera fácil y rápida la conexión de una aplicación HTML o flash con bases de datos MySQL, las más usadas del mundo, no solo por su versatilidad sino por su licencia gratuita.¹²

¹² José Lopez Quijado. Domine PHP y MySQL, Programación Dinámica en el lado del servidor. Alfaomega. 2008.

5. METODOLOGÍA

De acuerdo con el autor Roger Pressman¹³, el desarrollo de la aplicación propuesta fue abordado en tres grandes etapas: ANALISIS, DISEÑO E IMPLEMENTACIÓN.



El desarrollo de la aplicación multijugador tuvo como temática principal los carros de carreras acelerados por captura de sonido desde un micrófono. Y el público objetivo al que se dirige el juego son las personas entre los 10 y 26 años de edad residentes en Colombia.

Aprovechando la versión gratuita de SmartFox Server, se pueden conectar hasta 20 jugadores simultáneamente (Límite de la versión) en una sala principal de chat que se dividirá en diferentes salas de juego de 2 jugadores por sala.

Según el nivel de audio capturado por el micrófono de cada jugador sus automóviles acelerarán.

Las pruebas iniciales se hicieron localmente y más adelante se hizo el montaje de un servidor provisional para correr la aplicación en la Internet, con anchos de banda promedio.

A continuación se explicará detalladamente cada una de las etapas de desarrollo de la aplicación: El análisis, el diseño de la estructura, el diseño gráfico y por último la implementación del juego.

¹³ Roger Presuman. Ingeniería de Software, Un enfoque práctico. McGraw Hill. 5ª edición. 2002.

5.1 ANÁLISIS

En la fase de análisis se revisaron aspectos importantes que requería el desarrollo de esta aplicación, como las herramientas a utilizar, las necesidades, el usuario estándar y por supuesto las funciones que debería llevar a cabo el juego.

Para dejar claras todos los aspectos que implica el desarrollo de una aplicación multijugador como la realizada, se formularon tres modelos básicos que llamaremos Conceptual, estructural y organizacional.

5.1.1. Modelo Conceptual. El modelo conceptual de la aplicación se formuló a partir de los requerimientos básicos del juego y del ambiente que debe comunicar a los usuarios del mismo.

5.1.1.1. Ambiente a comunicar. El ambiente que comunica el juego es un ambiente amigable, que busca sumergir al jugador en el papel de un corredor de autos que no se aceleran como los automóviles comunes, sino que se arrancan por medio de la voz emitida por su micrófono. Pistas de carreras, diferentes autos y corredores, son elementos que debe encontrar el usuario en su rol de corredor.

5.1.1.2 Requerimientos de la aplicación. Se requiere una aplicación web que permita jugar al usuario contra otros jugadores mediante el nivel de voz, para así, competir en carreras de autos dentro de un juego usando un micrófono. En cuanto a la delimitación del desarrollo, se hará una primera versión del juego conectando los 20 jugadores simultáneos permitidos por la versión gratuita de SmartFox, dando lugar a 10 carreras a la vez, de a dos jugadores por sala de juego.

5.1.2. Modelo estructural. El modelo Estructural de una aplicación resuelve interrogantes como ¿Qué hará la aplicación? ¿Qué herramientas se usarán? Puntos clave para que el desarrollo del juego no saque de consideración ningún aspecto relevante.

5.1.2.1. Funciones del juego. La aplicación integrará un sistema de captura de audio en el que el nivel de la voz capturada permita avanzar a un objeto vectorial (Automóvil) dentro de las carreras.

Además del modo de juego multijugador descrito anteriormente, la aplicación debe contar con una menú principal en la interfaz, el cual ofrezca las opciones importantes al usuario de manera rápida, unas instrucciones de juego, y por supuesto una lista de usuarios conectados. Una sala de chat es una excelente herramienta que se debe implementar para generar una interacción entre los usuarios conectados.

5.1.2.2 Herramientas a utilizar. El uso de un servidor socket como Smartfox server requiere la utilización de Action Script como lenguaje de programación principal, teniendo en cuenta que las librerías de Smartfox están escritas en este mismo lenguaje.

Después de identificar Flash como herramienta base de la construcción del juego multijugador, se debe considerar la integración del mismo con bases de datos en el ámbito de los usuarios, sesiones y almacenamiento de puntajes; por esta razón es muy importante la utilización de un segundo lenguaje de programación dentro del desarrollo. PHP ha sido la elección en esta ocasión para integrar datos dinámicos con la interfaz de la aplicación desarrollada, ya que es uno de los más usados y es que mejor soportan los hostings comunes.

5.1.3 Modelo Organizacional. Este modelo contiene el contenido de la aplicación, sus secciones y su respectiva categorización dentro de la interfaz a diseñar para el juego. También se hace una pequeña descripción de cada una de estas secciones.

5.1.3.1 Garaje. La zona de garaje es una sección en la cual el jugador puede visualizar el automóvil que posee actualmente. Se debe mostrar en una ventana aparte de la principal.

5.1.3.2 Puntajes propios. El puntaje propio se visualiza en una ventana aparte que imprime una lista de los últimos 10 puntajes obtenidos en las carreras jugadas, además del total de puntaje que tiene un jugador en específico. Esto con el objetivo de que el jugador visualice su progreso o los diferentes resultados en varias carreras.

5.1.3.3 Puntajes generales o Ranking. El Ranking es una lista de los jugadores ordenada de mayor a menor puntaje, el cual revela a todos los jugadores la posición en que se encuentra cada uno de ellos.

5.1.3.4 Creación de Salas de Juego. Cada Carrera estará conformada por dos jugadores y traducida en salas de juego, en esta versión del juego se podrán crear 10 salas diferentes de juego con 20 jugadores conectados y jugando simultáneamente atendiendo al límite de conexiones que ofrece la versión de prueba de SmartFox Server.

5.2 DISEÑO DE LA ESTRUCTURA

Es importante definir desde el comienzo la secuencia de navegación del usuario, y la manera cómo se dará a paso a la implementación de código por parte del programador, de esta forma, se agilizan los procesos de desarrollo de una aplicación, y por supuesto se deja en claro cómo responderá el sistema ante cualquier acción de los usuarios de la misma.

5.2.1 Casos de Uso. Muestran el comportamiento y la secuencia de funcionamiento de la aplicación desarrollada, además de su interacción y respuestas al usuario activo.¹⁴

Figura 8. Diagrama de Casos de Uso Lobby



Extensiones y alternativas:

a. En cualquier momento, el sistema falla.

1. El sistema repite el paso 1.

2ª. Contraseña del usuario incorrecta.

1 El sistema muestra un mensaje de error.

El sistema repite paso uno y 2 hasta que la información digitada sea correcta.

¹⁴ Craig Larman, UML and Patterns Book. Pearson Education Inc. 2004.

- **Registro de nuevos usuarios**

Figura 9. Casos de uso Registro de nuevos usuarios



Extensiones y alternativas:

- a. El sistema falla en cualquier momento.
 - 1. El sistema repite el paso 1.
- 3a. El correo del usuario ya se encuentra registrado.
 - 1. El sistema muestra un mensaje de error.
 - 1. El usuario repite el paso 3.

- **Chat en zona principal**

Figura 10. Casos de uso Chat en zona principal



Extensiones y alternativas:

- a. El sistema falla en cualquier momento.
 - 1. El sistema vuelve al área de Login.
- 2a. El usuario envía mensajes privados.
 - 1. El usuario escoge el usuario a quien escribirá.
 - 2. El sistema muestra un dialogo con un input text.
 - 3. El usuario escribe y envía el texto privado.

- **Carrera:**

Figura 11. Casos de uso Carrera



Extensiones y alternativas:

a. El sistema falla en cualquier momento.

1. El sistema vuelve al área de Login.

1ª. El usuario pulsa el botón SALIR antes de que llegue otro jugador.

1. El sistema saca al usuario del Juego y muestra la ventana principal de chat.

2ª. El usuario NO permite el uso del micrófono por parte de la aplicación.

1. El sistema arroja un diálogo que explica al usuario que sin micrófono no puede hacer uso de la aplicación.
2. El sistema repite el paso 1.

3ª. Alguno de los jugadores se retira del juego antes de terminar.

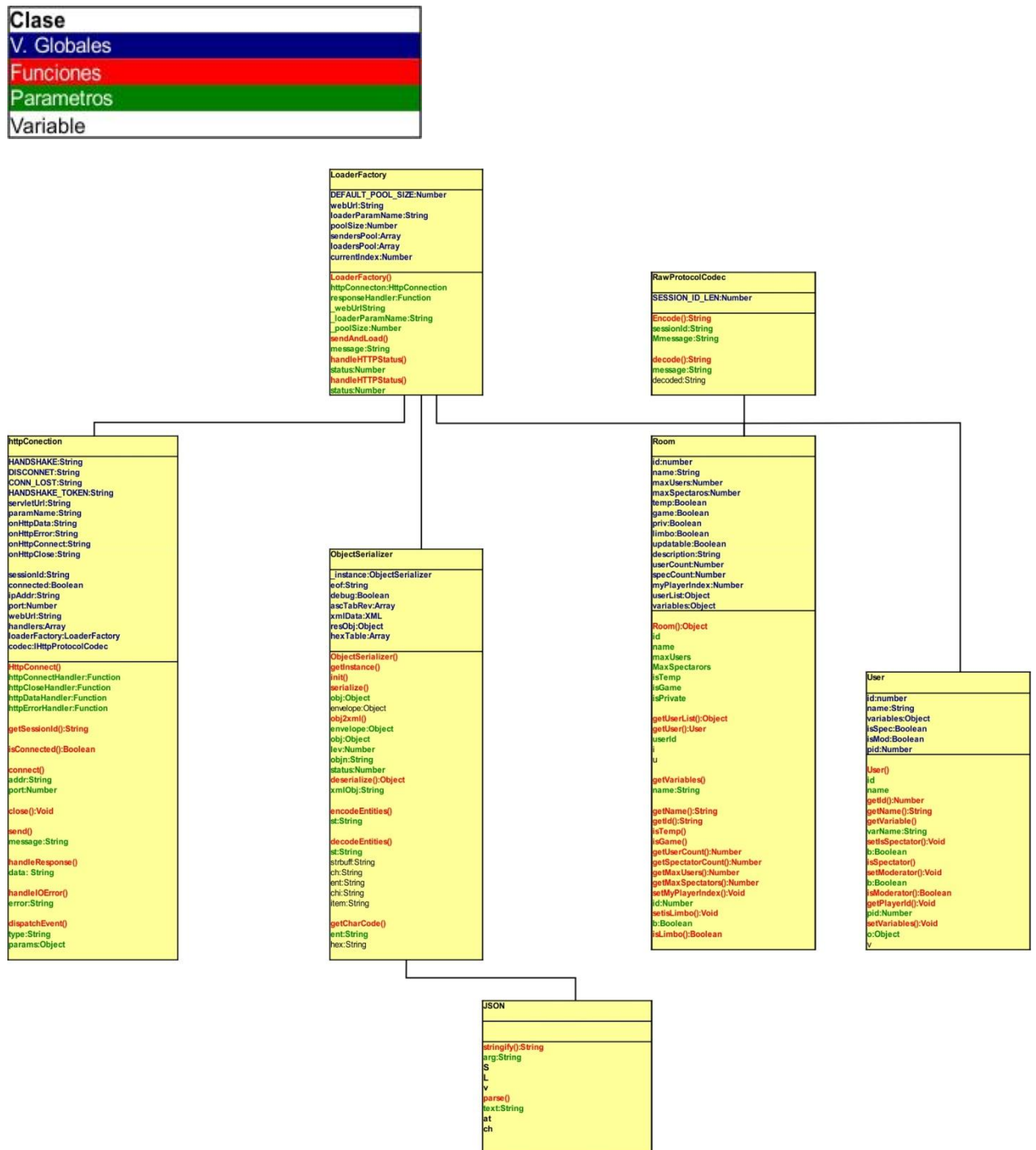
1. El sistema cancela la carrera y muestra un diálogo a los usuarios restantes de "ESPERAR OTRO JUGADOR"
2. Se repite paso 2 y 3 hasta que la carrera se de por finalizada.

5.2.2 Diagrama de Clases. Un diagrama de clases es un mapa estático que muestra la estructura de un proyecto de programación, mostrando sus clases, métodos, variables y atributos, en esta ocasión se desarrollaron tres diagramas de clases diferentes, uno para cada componente del proyecto. Un diagrama para las librerías de SmartFox Server, uno para la estructura PHP diseñada para el juego y un último diagrama para el archivo principal de la aplicación, el cual contiene funciones propias y actúa como iniciador de otros archivos.

En cuanto a la librería de Smartfox server, en mayoría, se usaron tres clases muy importantes: `httpConnection`, `Room` y `User`.

Entre las funciones que se usaron de `httpConnection`, se encuentran las dos funciones principales para abrir o cerrar conexiones de un archivo `.swf` con SmartFox: `connect()` y `close()`. Para ejecutar operaciones con salas de juego, se usaron funciones como `getUserList`, `GetUser`, `getVariables`, `getName`, entre otras, las cuales retornan información importante como variables activas en una sala, nombres de usuarios e IDs. E individualmente hablando de los usuarios, los métodos `getID` y `getName` fueron muy útiles para obtener información de los usuarios conectados a una sala o al chat principal. En la figura 12 se puede apreciar el diagrama de clases completo.

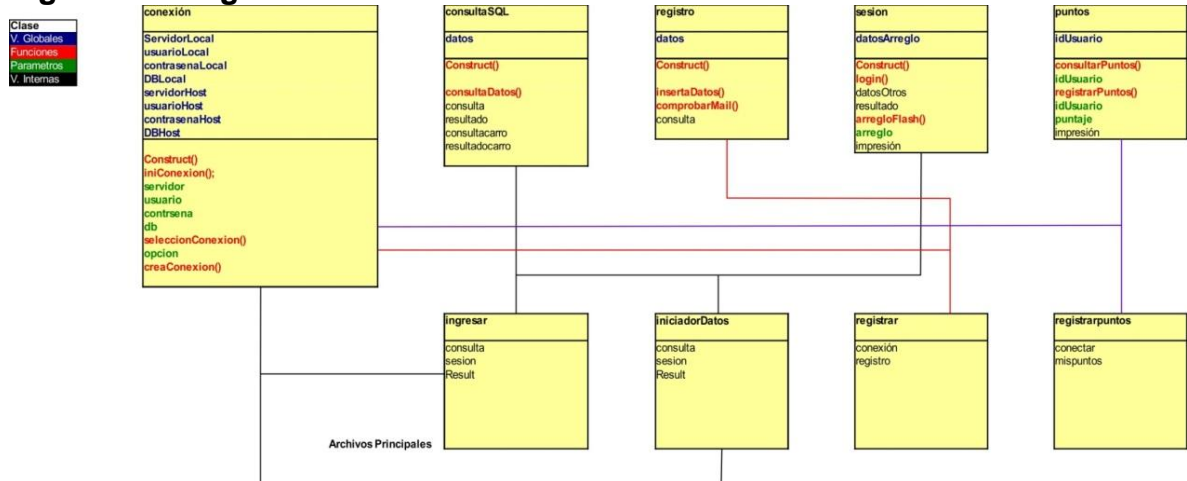
Figura 12. Diagrama de clases SmartFox Server



Fuente: Autor.

La integración de Flash con las bases de datos y otro tipo de operaciones del lado del servidor, requiere de programación en un lenguaje compatible como PHP. En este caso, se escribieron 5 clases que reunieron acciones como conexión a las tablas mysql de la base de datos diseñada, una clase para consultas SQL, una de registro de usuarios, otra de inicio y cierre de sesión y una última de registro de puntajes a los usuarios; cada una de ellas, con sus respectivos archivos de inicio como se puede observar en la Figura 13.

Figura 13. Diagrama de Clases PHP



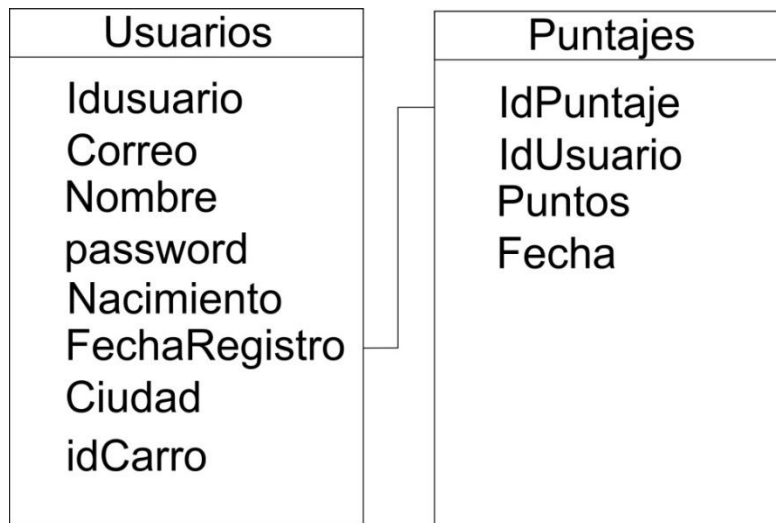
Fuente: Autor.

5.2.3 Estructuración de la base de datos. Al ver la información que debía almacenar la base de datos titulada 'MULTIPLAYER', fue estructurada de la siguiente manera:

Una tabla principal de usuarios, que contiene información básica como el nombre del usuario, correo electrónico, fecha de nacimiento y registro, contraseña y automóvil seleccionado actualmente.

Una tabla secundaria llamada puntajes, que contiene los puntajes obtenidos en cada juego por cada usuario, almacena datos como el ID del usuario, el ID del puntaje, la fecha del puntaje y los puntos.

Figura 14. Diagrama De La Base De Datos



Fuente: Autor.

5.3 DISEÑO GRÁFICO

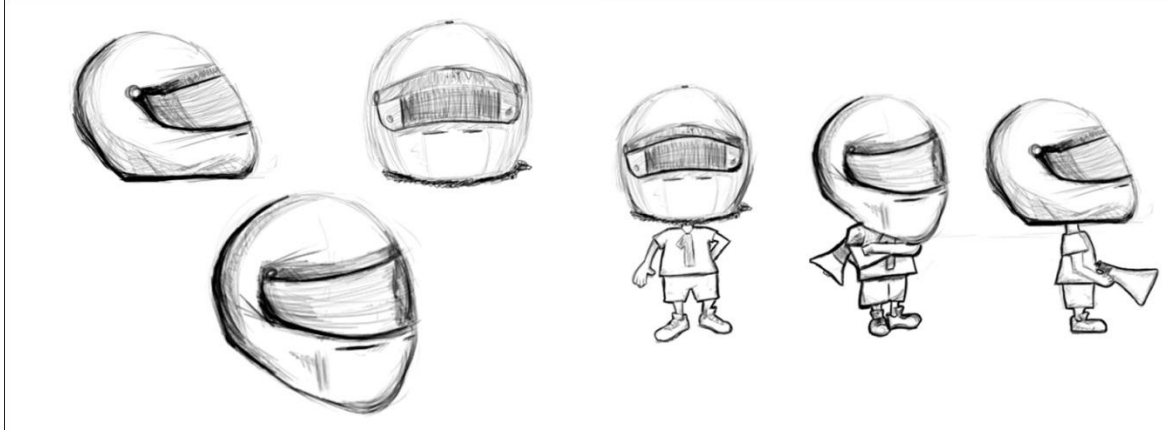
5.3.1 Nombre de la aplicación. El nombre de juego debía integrar dos componentes de manera clara y atractiva, el concepto de gritar frente a un micrófono, y las carreras de automóviles. La opción más adecuada desde el inicio fue la de titular la aplicación con el nombre de 'SCREAM RACING', decidiendo a la vez la tipografía que usaría el título en banners, logotipos, entre otros. La razón por la que se seleccionó la tipografía principal, fue el comunicar la caricatura y generar impacto y diversión con ayuda de las palabras que componen el nombre de la aplicación.

SCREAM RACING

TIPOGRAFÍA ESCOGIDA PARA LA IMAGEN DE SCREAM RACING

El personaje principal se realizó con la guía de uno de los ilustradores “retro” más importantes de esta generación (Christopher Hart), y conservando la apariencia de los cascos de corredores profesionales, se realizaron algunos bocetos, propuestas de color, y posterior vectorización del personaje:

Figura 15. Bocetos del casco y cuerpo del personaje



Fuente: Autor.

Figura 16. Personaje principal vectorizado



Fuente: Autor.

5.3.2 Logotipo. El proceso de diseño del Logotipo tuvo tres referentes básicos dentro del proceso de investigación:

5.3.2.1 Deportes Automovilísticos. Las Escuderías de diferentes Categorías como la Formula 1 y Kart ayudaron a definir el concepto de imagen de la aplicación a desarrollar. Colores fuertes, banderas a cuadros, cascos y bordes resaltados dejaron claro lo que debía comunicar un logotipo de un juego que involucraría al usuario en un ambiente automovilístico y a la vez caricaturesco.

5.3.2.2 Controles del juego. Se debía tener en cuenta que el principal control de los automóviles en el juego sería por medio de la voz del jugador, en combinación de esto con el nombre del juego (SCREAM RACING) se decidió incluir un elemento que represente la amplificación de sonidos, en este caso un megáfono.

Figura 17. Boceto del megáfono a incluir



Fuente: Autor.

5.3.2.3 Personaje Principal Diseñado. El personaje principal, un piloto con un casco grande en relación con su tamaño, debía ser parte activa, tanto del juego como de su imagen, por lo tanto se debía tener en cuenta la participación del mismo dentro del logotipo de la aplicación.

Usando estos tres aspectos, se dio paso al desarrollo de los bocetos, propuestas de color y vectorización del logo (Figuras 18, 19, 20 y 21)

Figura 18. Propuesta logo 1



Fuente: Autor.

Figura 19. Propuesta logo 2



Fuente: Autor.

Figura 20. Propuesta logo 3



Fuente: Autor.

La propuesta seleccionada para vectorizar fue la segunda, ya que es la que mejor deja en evidencia los tres aspectos resaltados anteriormente, y ofrece al público en general una idea mucho más clara de la temática del juego, además del objetivo que se tiene al jugarlo.

Figura 21. Propuesta seleccionada y vectorizada



Fuente: Autor.

5.3.3 Diseño de los automóviles. La selección de los dos automóviles de esta primera versión del juego se basó en tener un auto moderno y uno algo clásico, para así, ofrecer variedad al usuario y libertad de escoger; después de seleccionar los dos modelos a trabajar, se dio paso al proceso de caricaturización y vectorización de cada uno de ellos, siempre conservando la tendencia retro del juego.

Los modelos escogidos fueron el Chevelle 65, y el Ferrari 250 GTO, dos modelos de automóviles completamente distintos y que se prestaran para una caricaturización como la que se buscaba.

Figura 22. Automóviles escogidos y su respectiva caricatura

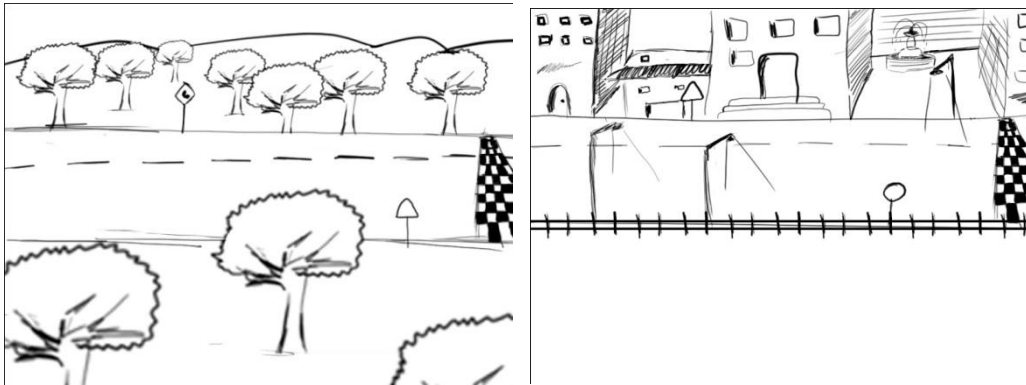




Fuente: Autor.

5.3.4 Escenarios. Se realizaron dos pistas de carreras diferentes, una que ilustró un escenario completamente rural y otro un poco más urbano, diferentes horas del día, pero la misma modalidad de juego.

Figura 23. Bocetos de pistas



Fuente: Autor.

Figura 24. Vectorización de pistas



Fuente: Autor.

5.3.5 Interfaz

5.3.5.1 Comportamiento. El comportamiento de una interfaz toca componentes como la consulta y navegación, en donde se incluyen preguntas como las siguientes:

- ¿Qué es éste producto y qué puedo encontrar?
- ¿Dónde me encuentro?
- ¿A dónde puedo ir?
- ¿Qué es lo más importante?

Basándose en estos interrogantes, se pudo iniciar a diagramar una interfaz capaz de responder positivamente a todos ellos y describir la aplicación por sí misma así: El producto es un juego de carreras en el que se encuentran varios jugadores reunidos. Puedo dirigirme a tres "lugares" diferentes dentro de la interfaz, como son el Lobby del chat, las ventanas de información como puntajes y garaje, y por último las carreras con otro jugador.

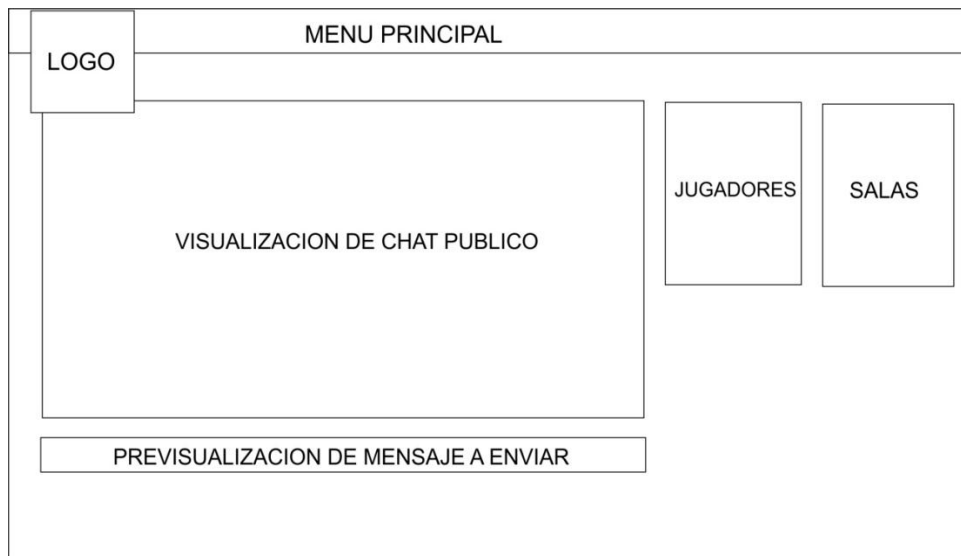
5.3.5.2 Presentación. El grado de excelencia de una interfaz depende fundamentalmente de su diagramación, la cual debe mostrar simplicidad, orden, y sobretodo consistencia, después es importante trabajar en su apariencia visual y por último en el contenido dinámico y estático de la aplicación.

- **Diagramación**

La creación de salas de juego debía ser parte del menú principal junto a otras opciones como "MI GARAJE" y el "SCREAM RANKING" (Un nombre homogéneo al título del juego para el ranking general de jugadores), además había que tener

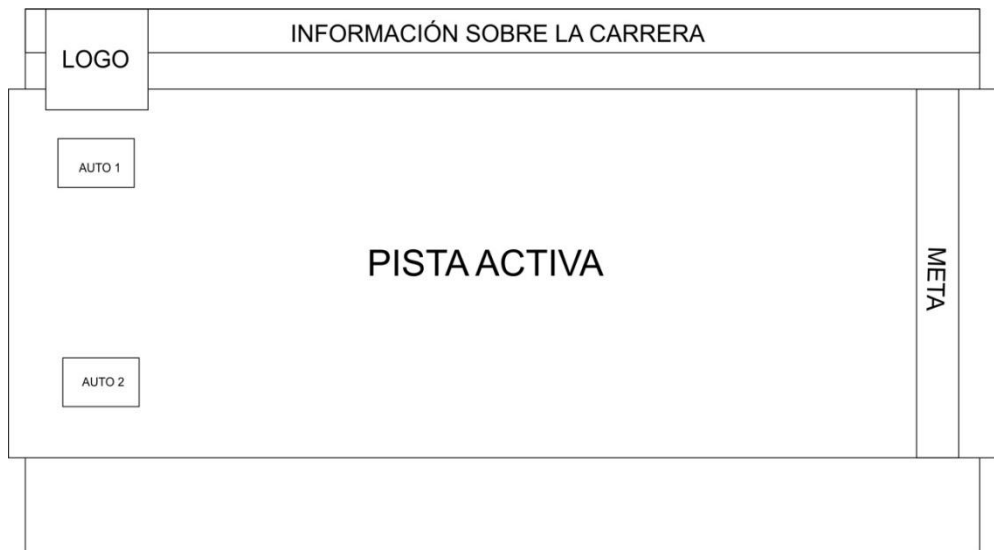
en cuenta el grado de importancia que se le quería dar al chat principal y a los espacios de jugadores y salas creadas.

Figura 25. Diagramación de la interfaz principal



Fuente: Autor.

Figura 26. Propuesta de diagramación de las carreras

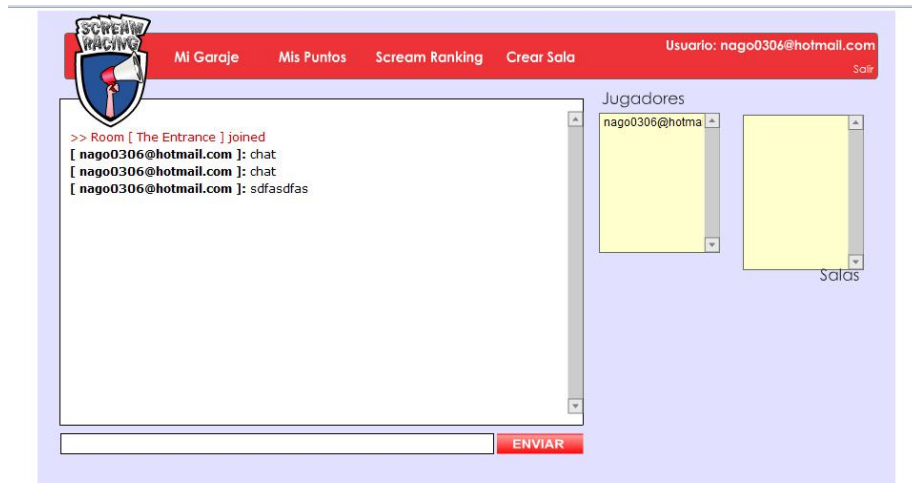


Fuente: Autor.

- **Apariencia visual**

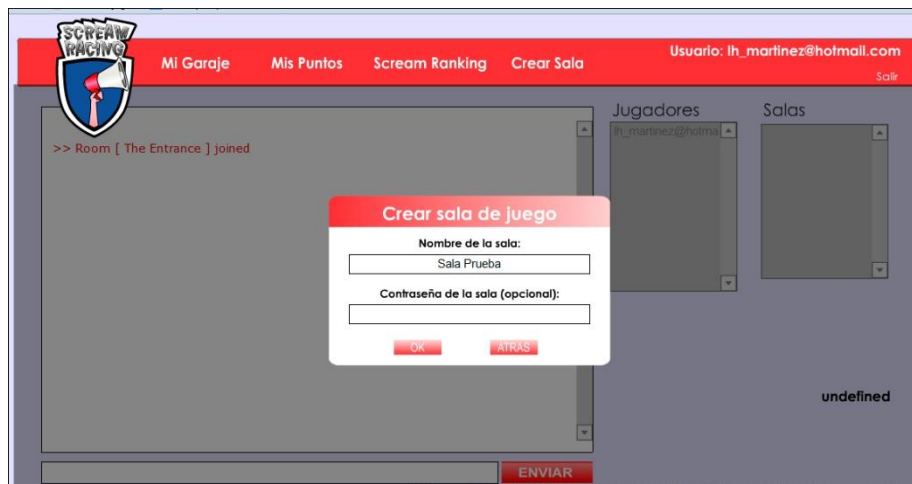
En este ítem lo más importante es considerar la audiencia de la aplicación o el público objetivo, que como se describió en la delimitación de este proyecto son las personas entre los 10 y 26 años de edad.

Figura 27. Apariencia de la interfaz principal



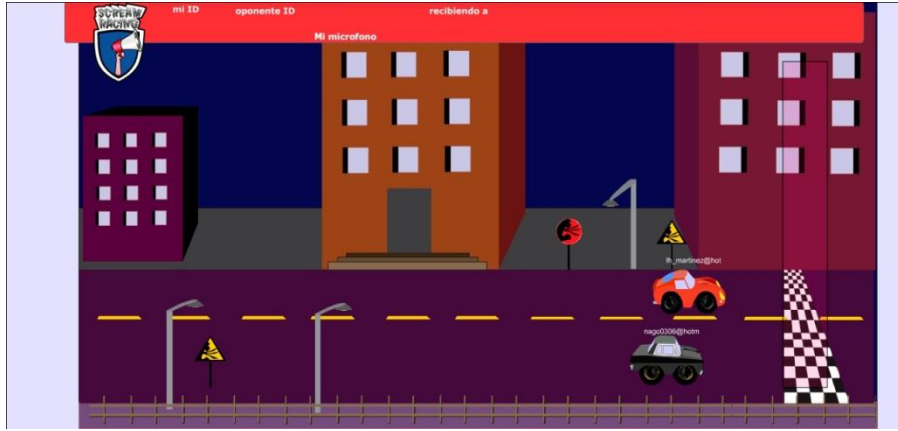
Fuente: Autor.

Figura 28. Diseño de las ventanas emergentes



Fuente: Autor.

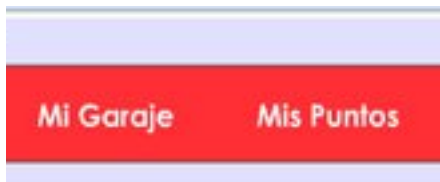
Figura 29. Apariencia de las carreras



Fuente: Autor.

- **Texto**

Además de la tipografía escogida para el logotipo, se seleccionaron dos fuentes secundarias, las cuales hacen parte de todo el contenido textual de la aplicación,



Century Gothic: Usada para las secciones del menú principal y los títulos importantes como JUGADORES, SALAS y los POP UPS



Arial: Usada para los mensajes y texto en general del juego.

5.3.5.3 Asistencia al usuario. Las instrucciones del juego se ubicaron al costado derecho del chat público, zona en la que son visibles y están al alcance de todos los usuarios de manera rápida.

Las Instrucciones que contiene la interfaz son las siguientes:

- Crea una sala de juego o ingresa a una sala que esté incompleta (un solo jugador)
- Si es una sala nueva, debes seleccionar una pista para correr, si no, solo activa tu micrófono y pulsa continuar.
- Alza la voz frente a tu micrófono para acelerar tu automóvil y ganar la carrera.
- Por cada carrera ganada conseguirás 50 puntos.

Además de las instrucciones básicas, cada menú posee su herramienta de ayuda el cual indica al usuario que puede hacer con cada uno de ellos.

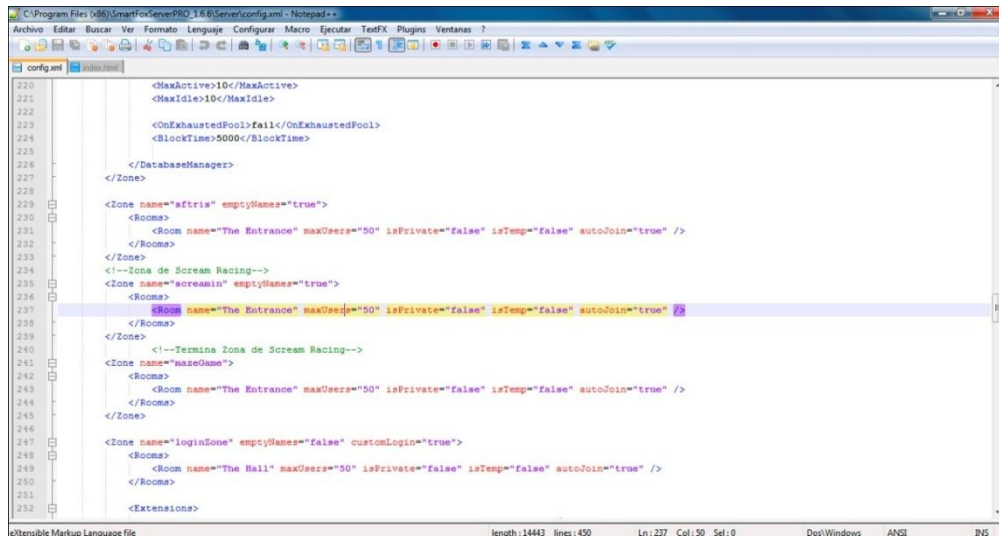
5.4 IMPLEMENTACION

5.4.1 Configuración de Smartfox Server local. A la hora de desarrollar juegos multijugador en Action script, Smartfox server funciona dividiéndose en zonas que pueden albergar una cantidad preestablecida de usuarios y salas de juego dentro de la aplicación que estemos desarrollando. También debemos tener en cuenta que la información de administrador se debe configurar desde el inicio para tener acceso al panel de control del servidor, el cual nos servirá más adelante para monitorear usuarios conectados, baneados, tráfico activo, zonas y variables de los juegos.

Todas las configuraciones anteriormente mencionadas son hechas después de instalado el server en nuestro ordenador, y antes de empezar a desarrollar la(s) aplicación(es) que deseamos, y son realizadas por medio de un archivo XML que contiene todo este tipo de variables llamado config.xml que pertenece a la instalación de Smartfox.

En esta oportunidad se creó dentro de config.xml una zona llamada 'screamin' la cual es la que contiene a los usuarios de la misma, y tiene la capacidad de transferir las variables de cada uno de ellos entre sí a la hora de jugar las carreras de 'SCREAM RACING'; también contiene una sala por defecto que se llama "THE ENTRANCE" la cual será útil para el chat principal del juego.

Figura 30. Screen shot de config.xml



Fuente: Autor.

5.4.2 La captura de audio. La clase de Action Script llamada MICROPHONE, hizo el proceso de implementación de este sistema mucho más sencillo.

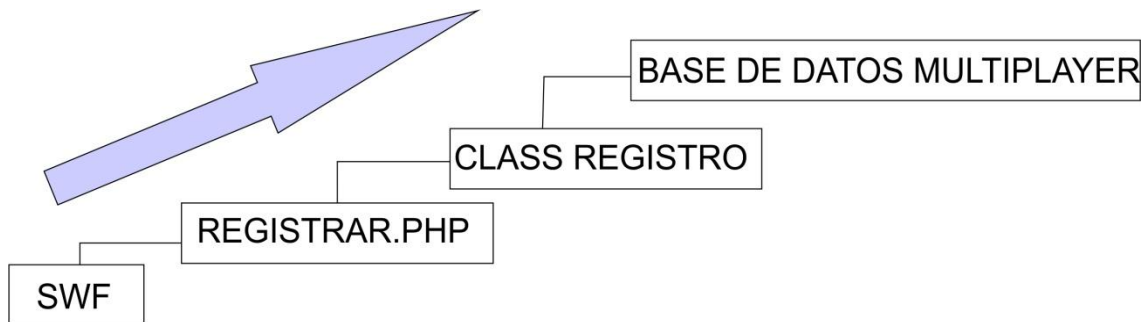
En esta etapa capturamos el nivel de voz que entra por el micrófono del ordenador en una variable X, y , trabajamos con una variable para el avance A, la cual es directamente proporcional a X. También se creó una tercera variable y dentro de este sistema, la cual está dada por el nivel de dificultad. Así, la aceleración de los automóviles está dada por la fórmula:

$$A = X/Y$$

5.4.3 Registro de jugadores nuevos y puntajes. Este proceso se desarrolla desde tres entornos diferentes, Action Script, PHP y MySQL.

Primero, desde Action Script enviamos a los archivos PHP las variables que necesitamos, en este caso, el nombre, correo electrónico, contraseña y demás datos a registrar del usuario, ejecutando un archivo REGISTRAR.PHP, el cual se basa en una clase que previamente se definió con nombre REGISTRO, así, ingresamos todos estos datos en la tabla USUARIOS de la base de datos MySQL.

Figura 31. Diagrama de funcionamiento del registro de usuarios



Fuente: Autor.

De la misma forma, el registro de puntajes se hace gracias a un archivo principal llamado REGISTRARPUNTOS.PHP que usa la clase PUNTOS para ingresar los puntajes obtenidos por determinado jugador en la tabla de puntajes con el ID del mismo.

5.4.4 Ingreso de jugadores registrados. En el momento en que determinado usuario pulsa ENTER para ingresar a su cuenta, ejecuta una función Action Script llamada LOGIN, la cual envía el correo y la contraseña digitadas, para que PHP valide si el usuario está registrado y digitó la contraseña correcta; si es así el archivo SWF avanza a la zona de Chat, de lo contrario el sistema reporta al usuario que existe un problema con la información ingresada.

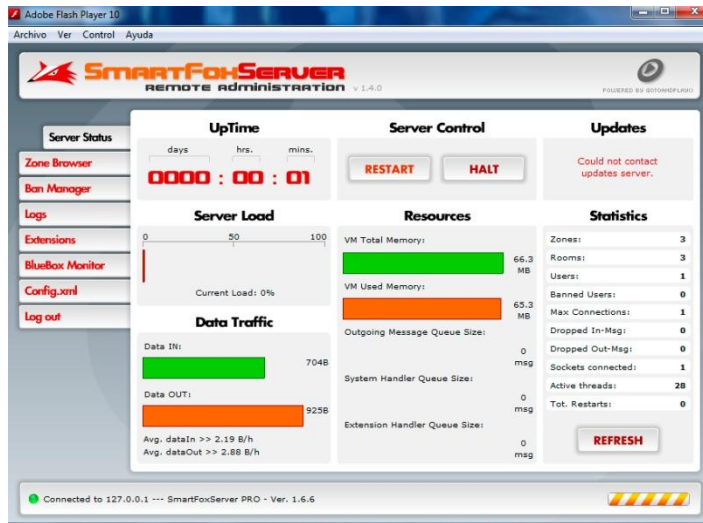
En este proceso no solo se ven involucradas clases definidas por mí, sino también se usan algunas funciones nativas de Smartfox, como la onlogin() que ingresa al server que un usuario está adentro, su zona, juego y variables propias.

5.4.5 Creación de salas de chat. La sala de chat principal del juego es una sala más de juego, que fue definida por defecto desde la configuración de la zona 'SCREAMIN' del SmartFox Server, cuando una persona entra a una carrera, sale de la sala de chat y entra en una distinta.

El campo del chat público se actualiza dentro de un mismo texto dinámico que va agregando cada uno de los mensajes que envían los usuarios activos.

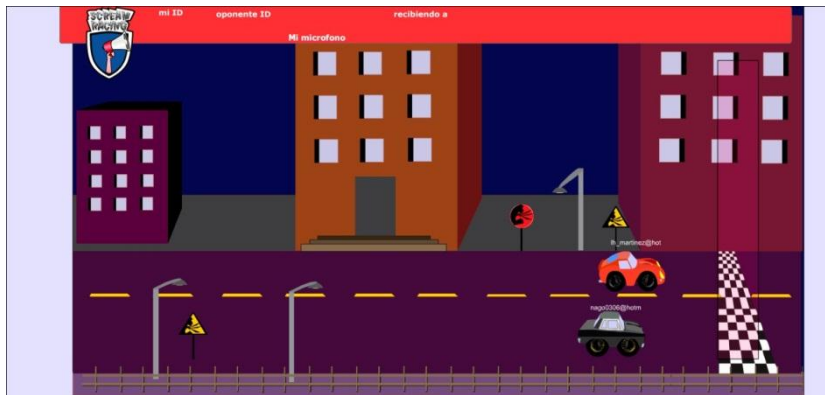
5.4.5 Creación de salas de juego y carreras multijugador. Cada carrera está dada por una sala de juego distinta, la cual incluye máximo dos jugadores, cada uno con sus variables propias, como son el automóvil que poseen, su velocidad, su nombre, entre otros. Variables que se actualizan 24 veces por segundo (FPS en flash) a ambos lados, para evitar el desfase entre un entorno y el del rival.

Figura 32. Screen shot del panel de control de smartfox



Fuente: SmartFox Cpanel.

Figura 33. Screen shot de una carrera en juego



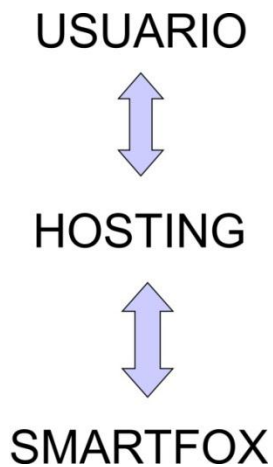
Fuente: Autor.

5.4.6 Instalación de Smartfox en servidor real e IP fija. Un computador estándar de escritorio fue dedicado exclusivamente al montaje del servidor Smartfox, para poder correr el juego online; en éste, se procedió a hacer la instalación corriente de SmartFox Server como en la etapa local, igualmente la configuración se hizo reemplazando el archivo config.xml con el config.xml que ya se tenía trabajado antes, solo fue modificada la etiqueta de IP del administrador, la cuál se reemplazó por la IP fija solicitada en a mi proveedor de Internet, ya que para este tipo de procesos requerimos de una IP fija, a donde deba apuntar la aplicación flash para conectarse con SmartFox.

De la misma forma, en la aplicación flash se reemplazó la conocida 127.0.0.1 local por la IP fija del servidor.

En un hosting común, se subió el resto de aplicación, es decir, los archivos PHP, SWF, javascript y demás componentes necesarios para correr el juego; así, el tiempo que lleva instalar apache, mysql, php y demás en un servidor, lo ahorré usando un hosting pago con las características necesarias para correr Swfs y PHP 5. Es decir, que desde este hosting mencionado la aplicación apunta al servidor SmartFox que se usó en casa.

Figura 34. Diagrama de funcionamiento usuario-hosting-smartfox



Fuente: Autor.

6. EVALUACION DEL PRODUCTO

Después de desarrollada una aplicación, se debe proceder a sesiones en las cuales se pone a prueba la aplicación en términos de funcionamiento y usabilidad, teniendo en cuenta que la mayoría de problemas técnicos y de comprensión de la aplicación pueden ser identificados de manera oportuna antes de culminar la realización de la misma, y por consiguiente, lanzarla al mercado. Pero ¿Cómo desarrollar unas buenas sesiones de pruebas? Quizás este es uno de los problemas más comunes en la etapa de evaluación de una interfaz o una aplicación, ya que en vez de permitir identificar errores, nos genera más; por esta razón es importante conocer el segmento de público al que va dirigido nuestro desarrollo y por supuesto plantear de manera correcta las encuestas que se harán.

A diferencia de las campañas publicitarias o la introducción de un producto a un mercado en específico, la etapa de evaluación de un diseño o aplicación no requiere de una muestra proporcional a la población total de enfoque. Por el contrario, entre más usuarios prueban la aplicación en una misma sesión, menos aprendemos de sus comportamientos o inconvenientes durante el test.

Jakob Nielsen¹⁵ afirma que es suficiente hacer las pruebas requeridas con 5 usuarios que estén enmarcados dentro del perfil de nuestro segmento de mercado; lo cual permite al desarrollador identificar casi en un 100% los problemas de funcionamiento y de usabilidad que pueda tener la aplicación. Es una cantidad pequeña, pero muy eficaz a la hora de identificar este tipo de errores, ya que después del primer usuario analizado, el test muestra comportamientos, errores, objeciones y peticiones similares a las del usuario anterior, por esto, entre más usuarios van probando la aplicación, son menos las observaciones nuevas que se obtienen, más tiempo empleado y por supuesto más costos.

En el momento de las primeras pruebas de usuario, se procedió a escoger 5 usuarios para las pruebas de la aplicación, asegurándonos que encajaran dentro del perfil del público objetivo definido desde el inicio del proyecto y que cumplieran con características de edad, ubicación geográfica, habilidades y nivel de uso de internet requeridas:

Cada una de las cinco personas tiene acceso a Internet desde su hogar, cumplió con el intervalo de edad propuesto entre los 15 y 26 años, todos residentes en Colombia y poseedores de por lo menos una cuenta de correo electrónico y dos perfiles en redes sociales o blogs como Twitter, wordpress, linkedin o Facebook, convirtiéndolos en usuarios básicos de Internet y a la vez haciendo de la muestra escogida, una muestra totalmente significativa para las pruebas de funcionalidad y

¹⁵ NIELSEN, Jakob. Jakob Nielsen AlertBox. ¿Why you only need to test with 5 users? 2000

usabilidad que debían hacerse en dos ocasiones: La primera para identificar una gran cantidad de problemas difíciles de detectar sin los usuarios, y una segunda sesión de pruebas después de que los errores encontrados en la primera fueran corregidos en su mayoría.

6.1. PRUEBAS DE FUNCIONALIDAD

En Primer lugar, es importante detectar qué mejoras de funcionamiento es importante hacer en la aplicación que se encuentra en desarrollo, así, podemos asegurarnos que la facilidad de navegación y comprensión que se ofrezca al usuario en la interfaz gráfica no se verá afectada por errores de programación o planeación. El método de Flavián, Guinalíu y Gurrea contiene dos puntos esenciales¹⁶: El primero solicita al evaluador mencionar y describir cada uno de los problemas de funcionamiento o navegación que se le presentaron en el tiempo que usó la aplicación; y el segundo punto pide al usuario dar el grado de repetición de 1 a 5, con que le ocurrieron los inconvenientes mencionados. Componiendose de una pregunta abierta y una cerrada, este test se convierte en pieza esencial para identificar problemas de funcionalidad, razón por la cual, a las personas de la muestra elegida se les aplicó un cuestionario como el siguiente:

- **TEST DE FUNCIONAMIENTO, SCREAM RACING 1.0**

Enumere los problemas de navegación y/o funcionamiento que se le presentaron durante la realización de la prueba y describa de 1 a 5 la frecuencia con que le sucedieron.

Este test tiene como objetivo principal identificar cuáles son los problemas de navegación o programación que tenía el juego y la frecuencia con la que sucedían este tipo de inconvenientes a cada usuario. Así se dio paso a su posterior corrección.

¹⁶ Flavián, Guinalíu, Gurrea. Métodos para la medición de Usabilidad. [Diapositivas] www.guinaliu.net 2010. 36 Diapositivas, color.

6.2 PRUEBAS DE USABILIDAD

Flavián, Guinalú y Gurrea definen la usabilidad como “La capacidad de un software de ser comprendido, aprendido, usado y se atractivo para el usuario en condiciones específicas de uso”.¹⁷

En cifras, una aplicación usable puede retornar la inversión de desarrollo hasta en un 800%, además reduce notablemente los costos de desarrollo, soporte y entrenamiento del usuario. Teniendo en cuenta la gran importancia de la usabilidad de cualquier aplicación que se desarrolle, existen pruebas y cuestionarios que permiten medir el nivel de usabilidad que poseen nuestras aplicaciones, además de identificar problemas graves de diseño e implementación.

- **TEST DE USABILIDAD, SCREAM RACING 1.0**

Nombre: _____

Edad: _____

email: _____

1. Totalmente en Desacuerdo
2. En desacuerdo
3. Ni de acuerdo ni desacuerdo
4. De acuerdo
5. Totalmente de Acuerdo

Escriba el número respectivo a la respuesta que considere adecuada para cada una de las afirmaciones:

- ☐ En la aplicación todo es fácilmente comprensible. _____
- ☐ Es sencillo utilizar la aplicación incluso la primera vez. _____
- ☐ Es sencillo encontrar las herramientas que busco en esta aplicación. _____
- ☐ La estructura de contenidos de esta aplicación es y fácilmente comprensible. _____
- ☐ Es fácil moverse por esta aplicación. _____
- ☐ La organización del contenido de esta aplicación me permite saber en qué lugar estoy y navegar por sus secciones. _____

¹⁷ Ibid., Diapositiva 4.

☐ Cuando Uso esta aplicación, siento que controlo lo que puedo hacer. ____

OBSERVACIONES:

En el caso de este cuestionario, permitía no solo observar el grado de sencillez y comprensión de los usuarios en cuanto a la aplicación se refiere, sino también recibir sugerencias, inquietudes y/o necesidades que tiene la muestra como usuarios medios del juego desarrollado.

En cuanto a la decisión de aplicar este tipo de cuestionarios a la muestra de usuarios escogida, se eligieron por algunas razones concretas como su brevedad y la variación de tipo de información que solicitan, ya que como sus mismos autores lo proponen, la información no debe ser 100% cualitativa ni 100% cuantitativa en ninguna encuesta o cuestionario que busque buenos resultados, ya que cada uno de estos aspectos complementa la intención del otro; por ejemplo no podemos saber si un error de controles se presenta continuamente en la aplicación o si fue una falla al dar instrucciones de juego, si no le solicitamos al encuestado un valor de 1 a 5 al grado de repetición de los inconvenientes que se le presentaron a lo largo de las pruebas.

También es importante resaltar que después de la primera evaluación por parte de los usuarios y la corrección de los problemas presentados en esta sesión, se realizó una segunda prueba de usuarios con la muestra, en donde se identificaron detalles y mejoras de la interfaz y de la aplicación en general que se describirán en el capítulo 7 de éste informe.

7. RESULTADOS

7.1. RESULTADOS DE LAS PRUEBAS

Antes de entretener y lograr la interacción entre los diferentes usuarios de la aplicación, uno de los objetivos principales para el diseño de la misma, es que las personas que lo usen puedan encontrar rápidamente la herramienta o información que buscan, y así puedan sentir el control de lo que están haciendo en el juego; sin embargo lograr esta meta, no sólo depende de su grado de usabilidad y las mejoras que se hagan en cuanto a ella, sino también de que la aplicación esté libre de fallas técnicas y de funcionamiento.

El promedio de los puntajes dados por los cinco usuarios en la primera jornada de pruebas, fue de 2.46 puntos sobre 5.0. Cifra que permitió enfocarse en algunas necesidades básicas del usuario frente al juego.

Dentro de las observaciones importantes que se lograron en las pruebas realizadas con la muestra de cinco usuarios, están la falta de explicaciones del modo de juego y la poca lectura que hacen los jugadores de las ventanas de dialogo del sistema, las cuales intentan dar una noción al usuario de la actividad que está realizando en un momento determinado, pero que a su vez, no cumplían con su objetivo, ya fuera por razones como tamaños de letra, ubicación, entre otras.

El área de chat proporciona un ambiente de interacción y amistad que al darse genera una mejor persuasión a la creación de salas y exploración del juego. Y entre muchos otros aspectos observados, se dio paso a crear una lista dividida en correcciones y mejoras para una próxima versión del juego. Cabe resaltar que después de haber implementado las mejoras necesarias, los puntajes de los usuarios en una segunda jornada de pruebas subió en promedio 1.5 puntos, llegando a obtener un resultado total de 3.96 puntos sobre 5.0 calificables, una mejora de más del 50% en funcionalidad y usabilidad de la aplicación.

7.2 CAMBIOS EN BASE A PRUEBAS

Teniendo en cuenta los resultados de los cuestionarios aplicados a los usuarios, se identificaron más de 20 items fundamentales para fortalecer diferentes aspectos de la aplicación desarrollada, además de satisfacer las necesidades de un usuario común de la aplicación; éstos se dividieron en dos categorías, correcciones inmediatas y mejoras que se proponen para una segunda versión del juego.

7.2.1 Correcciones Inmediatas. Estas correcciones se hicieron después de identificarlas en los cuestionarios de usabilidad y funcionalidad, con el objetivo de obtener resultados más altos en las pruebas aplicadas a la muestra de usuarios.

- **Cubrir contraseñas del formulario de registro:** Se optó por cubrir las contraseñas con asteriscos en el formulario de registro de usuarios
- **Claridad de los campos de texto de correo y contraseña:** Se reorganizó la posición de los campos de texto, Email y Contraseña, se hizo más visible el botón de registro ubicándolo bajo el logotipo del juego y cambiando su apariencia;
- **Escritura del carácter '@' en todos los navegadores:** Fue necesario retirar las etiquetas wmode del cuerpo HTML, las cuales ocasionan conflicto entre los caracteres especiales y los navegadores.
- **Al entrar a una sala ya creada, no es lo suficientemente visible el botón 'continuar':** Se cambió la apariencia y la posición del botón 'continuar', se agregó un aviso que comunica al usuario que está entrando en una sala de juego.
- **Escuchar un sonido que indica el arranque:** Se implementó una función que reproduce un sonido en el momento de iniciar una carrera.
- **Sincronización del arranque de cada carrera:** Se cerró el flujo de variables dentro de la sala cada vez que una carrera termina.
- **Conexión sin interrupciones dentro de la aplicación:** Se Aumentó el tamaño límite de los paquetes de datos en el xml de smartfox server.
- **Las herramientas de ranking y puntos deben fluir correctamente sin requerir desbloquear las pop ups del navegador:** Se cambió la modalidad de pop-ups por un javascript que muestra información dentro de la misma ventana de la aplicación.
- **Información de la carrera terminada con datos como el nivel de voz, puntuación y velocidad:** Se decidió imprimir más variables en la ventana de dialogo final de las carreras.

7.2.2 Mejoras para la segunda versión. Pensando en una segunda versión de la aplicación desarrollada, se debe buscar que el juego ofrezca amplias maneras de explotarse y jugarse en cuanto al papel de usuario se refiere.

A continuación se enumeran algunas de las mejoras que lograrían ampliar la experiencia de usuario dentro de esta aplicación, ofreciéndole a las personas libertad de juego y mayor control de las herramientas:

- Implementación una zona en la que los jugadores se enteren de su rival antes de competir y puedan configurar los atributos de su sala antes de iniciar una carrera.
- Es importante ingresar una mayor diversidad de automóviles que pueda seleccionar el usuario.
- La personalización de personajes es atractiva para los usuarios e importante implementar en una próxima versión. También cabe diferenciar el género femenino y masculino en los personajes del juego.
- Implementación de la opción de 'Recordar Datos' para mantener la sesión del usuario.
- Aumento moderado de la ilustración y animación de objetos de la interfaz y presentación.
- Implementación de un modo de juego individual que a su vez desempeñe el papel de entrenamiento (instrucciones del juego).
- Es viable permitir al creador de una sala de juego, configurar la cantidad deseada de usuarios a competir en una carrera.
- Es fundamental mostrar en más detalle el automóvil seleccionado dentro de la herramienta de garaje, se pasaría a considerar el grado de personalización de cada automóvil.
- Se deben incluir diferentes modos de juego multijugador, atributo configurable en la zona de espera de cada sala de juego.

7.3 RESULTADOS DEL DESARROLLO

Luego de las pruebas pertinentes, y la serie de correcciones necesarias para la primera versión de 'Scream Racing', se obtuvo una aplicación con gran similitud a la propuesta desde el inicio de este proyecto, con una mejora notable en la usabilidad y funcionalidad de su interfaz, gracias a los resultados que arrojaron las primeras pruebas de usuario, resultados que mejoraron casi en un 50% en las segundas pruebas, teniendo en cuenta que las mejoras se implementaron de manera eficiente.

Por otro lado se obtuvo una velocidad de transferencia satisfactoria, logrando que los dos jugadores de cada sala pudieran visualizar al mismo tiempo la actualización del entorno multijugador, permitiendo así, una buena experiencia de juego; esto se puede apreciar claramente en el videoclip realizado con el propósito de mostrar las pantallas de dos jugadores en un fragmento de una carrera online que tuvieron la oportunidad de jugar (www.makingroup.net/carreras/video),

Después de algunas revisiones por parte del tutor y otros docentes, también se logró identificar que la aplicación desarrollada funciona correctamente en las redes con una configuración de seguridad simple, en la que los puertos estén abiertos y disponibles para aceptar conexiones salientes o entrantes como en la mayoría de hogares y establecimientos públicos con acceso a Internet.

Teniendo en cuenta que para el intercambio de información entre jugadores y servidor se necesita usar el puerto 9339 con una dirección IP única, esto se convierte en un requisito básico de uso en el juego. Redes con configuraciones especiales y estrictas como las corporativas o las institucionales presentan limitaciones para ejecutar la aplicación, las cuales son ajenas al desarrollo de la misma y fue posible documentarlas con un documento que se puede leer en http://makingroup.net/carreras/documentos/informatica_umng.pdf, sección en la cual se evidencia que al ser la red de la universidad una red con una única dirección IP pública que contiene una gran cantidad de equipos con IP interna y sin traducción de dirección de red (NAT), el acceso a la aplicación es limitado para este tipo de redes corporativas o institucionales. Este es el caso de otras redes de carácter corporativo que fueron probadas con el propósito de identificar el problema de conexión que se presenta entre el cliente y el servidor encargado de intercambiar información entre todos los clientes conectados al juego.

8. POSIBLES VÍAS DE COMERCIALIZACION DE LA APLICACION

Al pasar los últimos años podemos afirmar que las empresas deben darse cuenta de la importancia del Marketing en la Internet; pero el Marketing digital no es solo publicitar nuestros productos en el amplio cyber-espacio que posee el planeta hoy en día, sino lograr una interacción prácticamente directa con los Usuarios y consumidores de nuestros productos, y así como afirma Josep-Lluís de Gabriel i Eroles: “Un dato que me llama la atención del estudio de la AIMC (Asociación para la Investigación de medios de Comunicación), es que la mayoría de personas, en su hogar, se conectan varias veces al día, en contraste con los años en los que la red no era un hábito, sino un entretenimiento ocasional.”¹⁸

Tabla 1. Posibles vías de comercialización de la aplicación

Frecuencia de conexión	Cantidad de veces	Porcentaje
Varias veces al día	22,92	56.7%
Todos o casi todos los días	12,25	30,30%
Varias veces a la semana	2,9	7,20%
Una vez a la semana	547	1,40%
Un par de veces al mes	229	0,60%
Una vez al mes	61	0,20%
Menos de una vez al mes	88	0,20%
Nunca o prácticamente nunca	1110	2,70%
NS/NC	349	0,90%

Fuente: Asociación para la Investigación de Medios de Comunicación (AIMC). Navegantes en la red. FEBRERO 2009.

Por esto, una aplicación 2.0 es el camino ideal para penetrar en mercados inimaginables. Gracias al fácil acceso a Internet que tienen los clientes potenciales de nuestros productos y servicios. ¿Cómo? Teniendo nuestras marcas presentes en redes sociales y aplicaciones web que conserven la noción de Interacción entre diferentes usuarios de la red.

¹⁸ DE GABRIEL I EROLES , Joseph-Lluís. Internet marketing 2.0. Reverté. 2010.

Teniendo en cuenta que las empresas deben dejar de bombardear con repetidos mensajes e interrumpir con publicidad, para pasar a generar un impacto real, creando una relación que sea viable por las diferentes direcciones que puede tomar la comunicación en canales digitales como 'SCREAM RACING' , podemos destacar entre muchos otros, tres grandes caminos de comercialización.

8.1 EVENTOS ENTRE CENTROS COMERCIALES

Interacción entre varios jugadores que se encuentren en eventos de diferentes centros comerciales.

8.2 CONCURSOS VIA TELEFONICA EN TELEVISION

Los Concursos vía telefónica en programas de entretenimiento se convirtieron en una tradición de la televisión mundial; Scream Racing, al igual que otros juegos de su mismo estilo, tienen un espacio dentro de este tipo de concursos.

8.3 PAUTAS DE MARCAS AUTOMOTRICES

Al ser Scream Racing un juego de automóviles, las marcas automotrices son patrocinadores potenciales para incluir sus nuevos modelos de carros dentro de la aplicación. Así dar a conocer sus nuevos autos a los usuarios de Scream Racing.

CONCLUSIONES

- Es fundamental estudiar y experimentar con otro tipo de herramientas, las cuales pueden resultar útiles para que un usuario interactúe de forma distinta con una aplicación, en este caso la captura de audio y la aceleración de objetos por medio de la voz, lograron un gran impacto en el público que tuvo la oportunidad de probar una primera versión de *Scream Racing*.
- Las pruebas de usuario son parte activa dentro del proceso de diseño e implementación de cualquier aplicación, ya que nos permiten evaluar nuestro producto por etapas, e identificar conflictos que muchas veces, el desarrollador no logra identificar individualmente. En el caso específico de este proyecto, dos jornadas de pruebas de usuario ayudaron a que esta primera versión de la aplicación y su interfaz demostrara un grado alto de funcionamiento, usabilidad e interactividad.
- El desfase del entorno es imperceptible al momento de jugar '*Scream Racing*' de manera online. En el video alojado en www.makingroup.net/carreras/video se puede apreciar el fragmento de una carrera entre dos jugadores y la actualización de entorno de las pantallas de cada uno de ellos.
- La facilidad de personalización de una aplicación tiene gran importancia para los usuarios, esto pudo observarse dentro de las sugerencias más escritas por la muestra de usuarios escogida para las jornadas de prueba del juego. Para esto se proponen algunas mejoras que de implementarse en una segunda versión de '*Scream Racing*', crearían mayor fidelidad en los usuarios, además de visitas frecuentes al juego.

BIBLIOGRAFIA

ABDELKHALEK, Ahmed. Parallelization and performance of interactive.

ALARCON. José Manuel, Programación en JavaScript. Anaya multimedia. 2001. 1008 p.

ANTZ INC Newsletter. Escala de Likert, 2010.

BAXLEY, Bob. Making the web work; Designing effective web applications Sams, Review, 2002. 474 p.

CAÑADAS OSINKI, Isabel y **SANCHEZ BRUNO**, Alonso. Categorías de respuestas en escalas tipo Likert. Psicothema, 1998.631 p.

DE GABRIEL i EROLES. Josep-Lluís. Internet marketing 2.0. Reverté. 2010. 236 p.

Efficient Resource Management, for Game Server Hostil. Florida, 2008. IEEE Paper.

Estándares W3C. Web design applications. <http://www.w3c.es/estandares/>.

FLORIA CORTES, Alejandro. Recopilación de métodos de usabilidad. Universidad de Zaragoza, 2000.

HART, Christopher. How to draw new “Retro style” Characters. Watson-Guption Publications, 2005.

JIMENEZ VELEZ, Carlos Alberto. El juego. Nuevas miradas desde la neuropsicología. Magisterio, 2008. 186 p.

LARMAN, Craig. UML and Patterns Book. Pearson Education Inc. 2004.

LOPEZ QUIJANO, José. Domine PHP y MySQL, Programación dinámica en el lado del servidor. Alfa omega, 2008.

MEDINA, Juan David. Desarrollo de aplicación Online para manejo de clientes. Universidad Militar Nueva Granada, 2008.

MULTIPLAYER GAME SERVERS. University Of Toronto,2004. IEEE Article.

NIELSEN, Jakob, Jakob Nielsen AlertBox. ¿Why you only need to test with 5 users? 2.000. <http://useit.com/alertbox/20000319.html>.

OSFLASH. Red 5, Open Source Flash server. <http://osflash.org/red5>.

PRESUMAN, Roger. Ingeniería de software, Un enfoque práctico. McGraw Hill. 5ª Edc. 2002. 640 p.

ROSELL, Eugeni. Diseño de logotipos. Gili: 2000. 208 p.

SMARTFOX SERVER OFFICIAL SITE. Smartfox Showcase. <http://www.smartfoxserver.com/showcase>.

VALLE, José Guillermo. Arquitectura cliente servidor. Corporación Universitaria Minuto de Dios. Colombia, 2005.

YUE YANNG; **CHANGJJIA** Chen y **YISHUAI** Chen. The estudy of online games based on virtual social relationship. Beijing Jiaotong University 2009. Article.